

# ARTxAI: Explainable Artificial Intelligence Curates Deep Representation Learning for Artistic Images using Fuzzy Techniques

Javier Fumanal-Idocin, *Member, IEEE*, J. Andreu-Perez, *Senior Member, IEEE*, Oscar Cordón *Fellow, IEEE*, H. Hagraas, *Fellow, IEEE*, Humberto Bustince, *Fellow, IEEE*

**Abstract**—Automatic art analysis employs different image processing techniques to classify and categorize works of art. When working with artistic images, we need to take into account further considerations compared to classical image processing. This is because such artistic paintings change drastically depending on the author, the scene depicted, and their artistic style. This can result in features that perform very well in a given task but do not grasp the whole of the visual and symbolic information contained in a painting. In this paper, we show how the features obtained from different tasks in artistic image classification are suitable to solve other ones of similar nature. We present different methods to improve the generalization capabilities and performance of artistic classification systems. Furthermore, we propose an explainable artificial intelligence method to map known visual traits of an image with the features used by the deep learning model considering fuzzy rules. These rules show the patterns and variables that are relevant to solve each task and how effective is each of the patterns found. Our results show that compared to multi-task learning, our proposed context-aware features can achieve up to 19% more accurate results using the ResNet architecture and 3% when using ConvNext. We also show that some of the features used by these models can be more clearly correlated to visual traits in the original image than others.

**Index Terms**—Automatic art analysis, Fuzzy rules, Image classification, Fuzzy clustering, Explainable artificial intelligence, and Deep learning.

## I. INTRODUCTION

The digitization of numerous paintings and collections worldwide has made it possible to employ the popular computer vision techniques and image processing on artistic data [1]. One of the most promising topics in this direction is the automatic analysis of paintings, in which these techniques are applied in creative tasks historically performed in most galleries and museums. Some of these are author verification [2], style analysis [3], and restoration [4].

Artistic image processing was traditionally performed using hand-crafted or ad-hoc features [5]. It is also possible to use

Javier Fumanal-Idocin and Humberto Bustince are with the Departamento de Estadística, Informática y Matemáticas, Universidad Pública de Navarra, Campus de Arrosadía, 31006, Pamplona, Spain. emails: javier.fumanal@unavarra.es, bustince@unavarra.es

Oscar Cordón is with the Dept. of Computer Science and Artificial Intelligence and with Andalusian Research Institute “Data Science and Computational Intelligence” (DaSCI), University of Granada, 18071 Granada, Spain. email: ocordova@decsai.ugr.es

Javier Andreu-Perez and Hani Hagraas are with the School of Computer Science and Electronic Engineering, University of Essex, Colchester, United Kingdom

Javier Andreu-Perez is with Simbad2, Department of Computer Science, University of Jaen, Jaen, Spain

information-based measures as features to perform classification and clustering, where style and author identification is performed based on the complexity characterization of each painting [6]. However, the advent of deep learning and convolutional neural networks has made automatically extracted features very popular [7], [8], [9]. Usually, these models are pre-trained and then fine-tuned for each specific task [10], [11], [12]. This is especially important for the case of artistic images [13]. One of the actual limitations of these models is that human experts perform their analysis based not only on visual cues but also on their expertise and their knowledge of the historical context, other paintings, materials, etc. [14]. Adding contextual and historical information to visual cues has been studied to perform different classification tasks in artistic image analysis [15], [16], [17]. However, there is no standard procedure to extract the contextual information associated with each artistic work. Besides, sometimes the context is not encoded in well defined labels. When the information is not well structured, like in a textual commentary, it is also necessary to discriminate those parts relevant to the task.

One of the most popular approaches to encoding this kind of information is knowledge graphs [15], [16], [17]. A knowledge graph captures the relationships between different concepts and attributions using the structure of a network [18], [19]. Indeed, graphs are a popular form of representing information [20], and they have been used to solve a myriad of problems in different areas of knowledge, like computer science [21], [22], biology [23], and the social sciences [24], [25]. However, when using a knowledge graph, a continuous space representation must be constructed from the nodes in the graph. This process is usually performed using deep learning models like node2vec [26]. Another possibility consists of using multi-task learning, in which a set of different related tasks are trained together so that the information obtained from one is also used in the others [27].

Capturing a painting context is also useful for improving the features obtained with a convolutional neural network (CNN). ResNet50 [28] has proven to have good generalization capabilities when trained on the extensively used Imagenet dataset. This generalization capability is particularly important in tasks where there is a significant domain shift and when visual information must be interpreted correctly in order to detect abstract concepts in the image [29]. The focus of the current paper is to study how general the features used in an artistic image classification problem are and how useful they

are when applied in other similar tasks (i.e. how useful they are to develop transfer learning). Doing so, we also measure if the network is learning a trivial solution to solve the task instead of finding patterns that can be generalized to new observations outside the training set [30]. We also want to test if the features obtained from a black box model can be correlated to known characteristics in the original image.

In order to achieve these aims, we present different ways to obtain such features, using only visual cues of the image and when additional information is also available. We also propose a new way to represent the contextual embeddings from different paintings using fuzzy memberships that expands previous approaches in this sense [31], [32]. We shall study how the Fuzzy C-Means clustering algorithm [33] and an adapted version of a fuzzy-rule based fuzzy clustering algorithm can be used to construct an embedding space and how this embedding captures relevant information from the original texts. We shall also study the use of Contrastive Language-Image Pre-Training (CLIP) features [34], [35]. In order to map the deep features obtained with these models to known visual cues, we will employ approximate reasoning through the means of fuzzy rules.

This article’s **key innovations** are:

- A *novel* methodology that combines fuzzy clustering and multi-task learning enhances deep learning model performance in artistic image classification.
- Extracted deep features are interpreted via fuzzy rules based on semantic information from the original image for *the first time*. We also measure how good is each casual propositional relation as an explanation for each feature.
- *Explainable* comparisons between the painting styles of different authors by means of fuzzy rules are developed.

To our mind, this research constitutes an innovative application of fuzzy set theory since, to the best of our knowledge, i) fuzzy rules are applied to interpret deep features based on semantic information from the original image; ii) explainable comparisons between authors’ painting styles are provided based on fuzzy technologies; and iii) our approach outperforms other standard and nonfuzzy computational intelligence techniques in both tasks.

The rest of the paper is organized as follows: in Section II we recall some of the previous concepts required to understand this work and review some relevant literature. In Section III, we introduce the proposed framework for artistic image classification using contextual embeddings and the different methods proposed to obtain contextual embeddings from textual annotations. Subsequently, Section IV shows the results obtained using the different context-aware and non-context-aware methods. Then, in Section V, we discuss the experimental setup and describe our method to explain deep features. Finally, in Section VI, we give some final conclusions and future lines for this work.

## II. BACKGROUND

In this section, we will review some previous works regarding fuzzy clustering and fuzzy rule-based classification, representation learning, and artistic artwork classification.

### A. Fuzzy rule-based classification and fuzzy clustering

The fuzzy rule-based classification consists of discriminating observations into different categories using rules that follow this structure [36]:

$$\text{IF } \mathbf{x}_1 \text{ is } \mathbf{a}_{j1} \dots \mathbf{x}_n \text{ is } \mathbf{a}_{jn} \text{ THEN class } j \text{ for } j = 1, \dots, C \quad (1)$$

where  $\mathbf{x}$  is a multidimensional vector,  $j$  is the consequent class,  $\mathbf{a}_j$  is an antecedent linguistic value for class  $j$ , and  $C$  is the number of different classes. For this purpose, each attribute is re-escalated into the  $[0, 1]$  unit interval, and then, the  $n$  different attributes are partitioned into different fuzzy subpartitions.

There are different ways in which these fuzzy subpartitions can be generated [37]. There are also different algorithms to generate a set of fuzzy rules to classify the samples [38], [39]. It is also possible to use fuzzy rules to perform clustering [40]. Fuzzy rules have been very useful for explainable AI because they can be easily interpreted by human stakeholders [41].

Fuzzy C-Means (FCM) is a well known fuzzy clustering algorithm in which each element is assigned not only to one group but rather presents a fuzzy membership to each of the groups considered [33]. The algorithm randomly assigns a coefficient for each observation to each cluster. Then, it computes the centroid for each cluster and computes each membership again. The process is repeated until convergence. There is a need to provide a cluster number  $c$  as input to the method.

### B. Artistic artwork classification

Historically, automatic art analysis has been performed using handcrafted features that relied on color, brushwork, and scale-invariant features [5], [42]. Then, once the features were extracted, they were used to train different kinds of classifiers. The most popular tasks include identifying the author, the style, and the theme of a painting [31].

The advent of deep learning has substituted the use of manual features with automatically extracted ones [43]. These features have been extracted using different networks, like the Residual Network (ResNet) [28] and the VGG16 [44]. Pre-trained networks can be used to recognize different shapes and entities in images, and they have also been extensively used in art analysis. It is also possible to fine-tune these networks to those tasks [31].

To further study art from a semantic perspective, visual information can be combined with contextual information from the art pieces. There are different ways in which this information can be incorporated into the classification framework. One possibility is to train simultaneously different classification tasks in a multi-task setting [31]. In this way, features learned to classify one image can help learning in other tasks and *vice versa*. It is also possible to use knowledge graphs constructed from the dataset itself [16] or from external information [45].

In spite of these successes, some of these methods still present some shortcomings. It is difficult to tell when the predictions are based on meaningful artistic knowledge of

shortcuts. Besides, the categorization of specific parts of an image (like a cat, a dog, etc.) differs from aspects like an author, which cannot be necessarily correlated only to specific parts of the image. This makes some explainability models like Grad-CAM [46] less useful for this task.

### III. PROPOSED ARTISTIC IMAGE CLASSIFICATION FRAMEWORK

In this section, we shall discuss our proposed classification framework, illustrated in Fig. 1, and the different alternatives studied to train and finetune such a model. This model shall be capable of solving one task or various tasks at the same time (style, year, type, and author identification) at the same time.

#### A. Model architecture

Our proposed framework consists of two different parts (see Fig. 1). We used a ResNet50 [28] to extract visual features for each image. The last layer of the ResNet is substituted by a linear classifier with the appropriate number of classes as the output.

In order to train the model, we have used different loss strategies. The most basic strategy is to train the model using the features obtained from ResNet using the standard cross-entropy loss for all classes  $C$ :

$$l(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_{ci} \log \hat{y}_{ci} + (1 - y_{ci}) \log(1 - \hat{y}_{ci}) \quad (2)$$

In order to exploit all the information available in the training process for each task, we train a combination of losses in a multi-task learning (MTL) setting, fusing them in a single simultaneous training objective for the author, type, school, and timeframe. In that case, the classification loss is the average of the four different cross-entropies for each task:

$$l_{MTL} = \frac{1}{4} (l(y_{author}, \hat{y}_{author}) + l(y_{type}, \hat{y}_{type}) + l(y_{school}, \hat{y}_{school}) + l(y_{time}, \hat{y}_{time})) \quad (3)$$

#### B. Context extraction

In order to aid regularisation in the classification task, we use additional contextual information to condition the output from our model. This information is encoded using a real-valued vector, computed using one of the methods proposed:

- 1) Node2vec in a knowledge graph that connects the paintings according to their shared attributes [31].
- 2) Fuzzy memberships over the contextual annotations for each painting.
- 3) Features obtained with a CLIP autoencoder on the contextual annotations.

Concerning the second method, *fuzzy context encoding*, the idea of using fuzzy clustering for this task is that the space formed using a word embedding method can be a faithful representation of the original domain, but it might not be useful to solve the task at hand. Since we are interested in using

these features to discriminate between classes, we are more interested in the topology of the representation obtained and the groups that are naturally present in them.

We expect that these groups agglomerate categories that are not mutually exclusive. For example, in the case of artistic representation, style and year can be very correlated because of artistic movements. In this case, there are many more possible examples: landscapes can be grouped together but belong to different authors, etc. Fuzzy clustering is the most suitable clustering tool for this task since we intend to express the membership to different, not mutually exclusive groups. For each observation, we have a fuzzy membership degree for each pertinent group.

Besides, fuzzy clustering is much more convenient than the traditional K-means algorithm for this task since fuzzy memberships are all in the same  $[0, 1]$  range. This means that the resulting vector will be a real-valued vector with more information than a one-hot encoding corresponding to a unique cluster. In addition, memberships to clusters that are far away can be modeled with a 0 in both cases, while distances in the K-means can be very different in magnitude but represent the same thing: the observation is not in this cluster. Thus, fuzzy memberships are also more suitable than Euclidean distances in a hard clustering approach for this task.

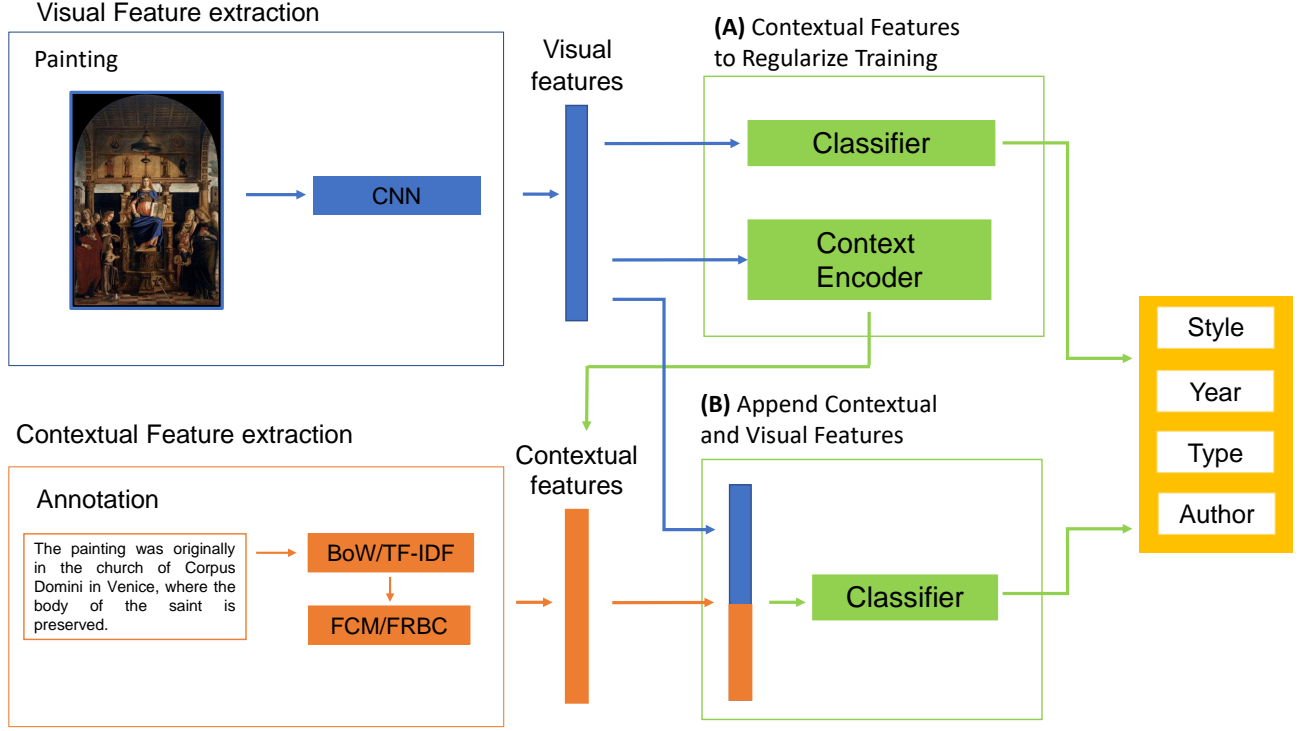
In order to compute the fuzzy memberships, we first encode the text annotations using Bag of Words (BoW) or TF-IDF encodings. Once this codification is computed, we run a fuzzy clustering algorithm to obtain the desired number of memberships.

The most popular fuzzy clustering algorithm for raw observations is the FCM [33] to obtain soft partitions of the feature domain. Fuzzy rules have also been intensively used to perform classification and data mining. However, many clustering algorithms are based on the direct optimization of a partitioning objective function without intermediate structures that explain the rationale of the decision for the optimal partition solution. Hence, we adopt the approach proposed by Mansoori et al. of integrating a fuzzy rule-based process inference in the clustering formation [40]. The modification of the original clustering algorithm includes two main changes:

- 1) The original algorithm gives, as a result, a crisp clustering. In order to return fuzzy memberships to the distinct groups, we use the value of the consequent for each rule selected in the algorithm.
- 2) The original algorithm had a stopping condition based on a stopping parameter so that when a percentage of the original data was assigned to a group, it ends its execution. Since there are no proper criteria to choose this parameter, we stop when all the original samples have been removed from the dataset.

The resulting algorithm is displayed in Algorithm 1.

This approach has an advantage over the FCM. In the FCM, the sum of all memberships must sum 1. This means that the bigger the contextual vector is, the lesser value each membership will be. In the case of fuzzy rule-based clustering, memberships for each cluster are independent, so there is no such restriction. In this way, it works as a possibilistic FCM algorithm [47]. Furthermore, we do not need to specify the



**Fig. 1: Scheme of the two proposed classification frameworks using contextual features.** Option (A) uses the contextual features to regularise the training of the CNN. The last layer of the network is replaced to fit the number of classes in the task. Option (B) appends the visual and contextual features in a single vector that is fed to a final classification layer.

---

**Algorithm 1:** Fuzzy clustering algorithm with fuzzy rules Algorithm

---

**Input :**  $X$ , being the set of main data observations:  
 $X \leftarrow \{\mathbf{x}_1, \dots, \mathbf{x}_m\}$

**Output:**  $C$ , being the cluster fuzzy memberships for each observation in  $X$ :  $C \leftarrow \{\mathbf{c}_1, \dots, \mathbf{c}_m\}$

$z \leftarrow 1$

$X_{og} \leftarrow X$

**repeat**

$X' \leftarrow$  a set of synthetic samples

$q \leftarrow X - \text{centroid}(X)$

$q' \leftarrow X' - \text{centroid}(X')$

**if**  $q' < q$  **then**

        Go to the start of the loop

$R \leftarrow$  set of rules to discriminate between  $X$  and  $X'$

$r \leftarrow$  rule with the highest average value for the antecedent

$X \leftarrow X \cup X'$

**for**  $i = 1, \dots, m$  **do**

$c_{1z} \leftarrow$  degree of truth  $r(X_i)$

    remove\_vector  $\leftarrow c_{*z} > 0.5$

$X \leftarrow X \setminus X_{\text{remove\_vector}}$

$z \leftarrow 1$

$X_{og} \leftarrow$  original samples in  $X$

**until**  $|X_{og}| = 0$ ;

---

number of clusters, as the process has a natural way to finish when all the observations have been assigned to a group.

### C. Context conditioning

Once we have computed the contextual vector, there are two different approaches to combine it with the visual cues:

- We use the contextual information vector in order to “regularise” the visual features. To do so, we have two “final” layers: one encoder that transforms the final feature vector of the network into the contextual features and another one that performs the classification. These encoders are single fully connected layers with a Rectified Linear Unit activation function (Fig. 1a). This is a trendy scheme in the literature to join information from heterogeneous sources [35].
- We append the contextual information vector to the visual characteristics vector. Then, we use a fully connected layer to learn from the resulting vector (Fig. 1b).

Given  $r$ , the final embedding obtained from the ResNet, and  $m$  the number of clusters obtained with the fuzzy clustering, the loss function for the reconstruction of the fuzzy membership vector is the Smooth L1:

$$\delta_{emb}(a, b) = \begin{cases} \frac{1}{2}(a - b), & \text{if } |a - b| \leq 1 \\ |a - b| - \frac{1}{2}, & \text{otherwise} \end{cases} \quad (4)$$

$$l_{emb} = \sum_{i=1}^n \sum_{j=1}^m \delta_{emb}(w_{ij}, r_{ij}) \quad (5)$$

where  $a, b \in \mathbb{R}$ ,  $w_{ij}$  is the  $j$ -th element of the fuzzy membership of the  $i$ -th sample,  $r_{ij}$  is the  $j$ -th element of the reconstructed contextual vector of the  $i$ -th sample.

Finally, we combine both the classification and the embedding loss using a convex combination of both, so the final loss is:

$$l = \alpha l_{class} + (1 - \alpha) l_{emb}, \quad (6)$$

where  $\alpha \in [0, 1]$ . Choosing the correct  $\alpha$  value is important in order to avoid one loss function “dominate” the other. We have chosen 0.9 as the value since it gave good results in the literature [16].

#### IV. EXPLAINING DEEP FEATURES USING FUZZY RULES

The interpretation of deep features in CNNs is still a challenging task, as they are the output of a large number of matrix operations. It is possible to grasp a better understanding of their behaviour if we correlate them to known characteristics in the original image.

In order to do so, we propose to use a FRBC that will map known features to the degree of activation of the deep features used to classify the paintings in each task. This will allow us to understand the predictions done by the network using abstract concepts and Grad-CAM heatmaps [46].

##### A. Extracting style information

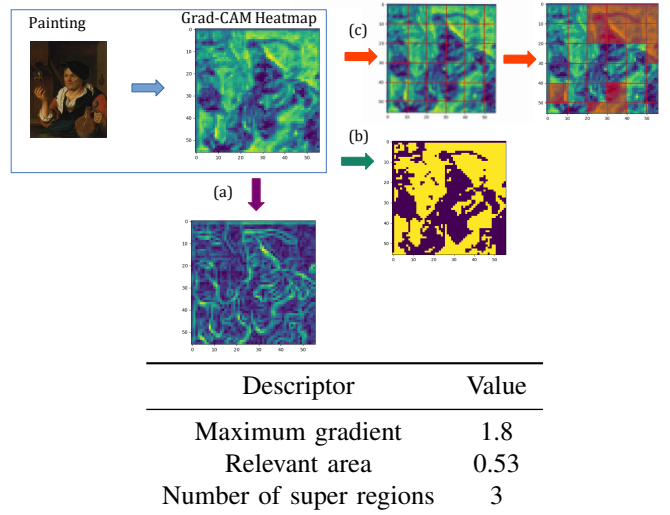
To extract known features, we first use another ResNet50 model trained to recognize artistic movements in a painting. The output of this model will be a vector containing the score for all possible styles considered. This model is trained on 100,000 paintings belonging to 2,300 different authors from the WikiArt dataset, which contains 27 different artistic styles. We trained the ResNet for a maximum of 300 epochs with a limit of 50 iterations without a tangible improvement in the model performance. It obtained a 53% accuracy overall (specific details about the datasets used can be found in Section V-A).

##### B. Characterizing visual focus

In order to join conceptual and visual concepts, we have studied the gradient maps of the SemArt models using Grad-CAM [46], which shows the regions which contributed significantly to the network prediction. Since we are studying four different tasks, we obtain for each image not one, but four different Grad-CAM heatmaps. To get overall information we fuse them using the average of those values. Once we reduced the different Grad-CAM maps to only one, we characterize each of them focusing on three different properties:

- 1) The percentage of the image with significant attention values.
- 2) The magnitude of the biggest gradient in the heatmap.
- 3) Explanation of connected parts in the heatmap.

In order to consider a pixel relevant, we just compare it against the average value for that image. Those bigger than the average are considered relevant.



**Fig. 2: Example of the extraction of the different descriptors for a Grad-CAM heatmap.** (a) shows the gradient magnitude for each pixel. We choose the highest value from that image as the descriptor. (b) shows the pixels denoted as relevant because their value was higher than the average value in the image. (c) shows the division into different regions and those designed as relevant. Then, we can generate the “super” regions. In the adjacent table, we can find the numeric values for each descriptor in this image.

**Table I: Statistics of the Grad-CAM heatmap descriptors for the SemArt dataset.**

Descriptor	Average value	Standard deviation
Maximum gradient	2.27	0.49
Relevant area	0.37	0.08
Number of super regions	2.90	1.39

In order to compute the maximum gradient in the image, we compute the Sobel filter, both in the horizontal and vertical axis. Once we have the gradient for each direction, we compute the gradient magnitude in each point from those vectors using the Pythagorean theorem. Then, we choose the biggest one as a result.

We divide the image into  $N$  squares of equal size to designate the number of connected components in the image. Then, we denote which of these regions was relevant in the classification. We designate as relevant those regions whose average value is bigger than the average value of the whole image. Then, we connect the regions regarded as relevant that are adjacent, which results in a series of “super” regions. The number of connected components obtained is the same as the number of the “super” regions formed in the image. Fig. 2 shows an example of the results for this characterization for one image.

Finally, Table I shows a summary of the statistics of these descriptors. We can see that the average value of the relevant parts of the image is about a third and that there is an average of 3 connected components in each painting.

### C. Mapping known features to deep features

In order to map interpretable patterns from the known features to others, we use a FRBC. Our aim in using the FRBC is to obtain an interpretable classifier. We shall obtain this using interpretable features and limiting the number of rules and their antecedents.

In order to obtain a reasonable-sized FRBC, we set 15 as the maximum number of rules and 4 as the maximum number of antecedents. However, the real number of rules used will be reduced from that number using a quality metric. We also choose three linguistic labels for the fuzzy partitions that are easily interpretable: low, medium, and high. To obtain the prediction for a sample, we compute the dominance score (DS) of each rule  $r$  [48]. This metric measures how often the rule fires and how its strength is compared to the remainder so that rules that are good in both senses are preferred.

The DS for each rule is the product of their support and confidence, with the support,  $s_r$  being [49]:

$$s_r = \frac{\sum_{\mathbf{x} \in Cons_r} w_r(\mathbf{x})}{|R|}, \quad (7)$$

where  $w_r(x)$  is firing strength of rule  $r$  for the sample  $x$ ,  $\mathbf{x} \in Cons_r$  is the set of observations whose ground-truth class corresponds to the rule class consequent and  $R$  is the set of all rules in the FRBC. The firing strength of the rule is the product of the truth degrees of all antecedents of the rule (product t-norm). Confidence is defined as:

$$c_r = \frac{\sum_{\mathbf{x} \in Cons_r} w_r(\mathbf{x})}{\sum_{r'=1, \mathbf{x} \in Cons_{r'}} w_{r'}(\mathbf{x})} \quad (8)$$

So, the DS of each rule,  $ds_r$ , is defined as:

$$ds_r = s_r * c_r \quad (9)$$

Finally, we compute the association degree,  $as_r(x)$ , using  $w_r(\mathbf{x})$  and  $ds_r$ :

$$as_r(x) = w_r(\mathbf{x}) * ds_r. \quad (10)$$

Each sample is classified according to the consequent class of the rule with the highest association degree for that sample:

$$P(x) = Cons_{arg \max (as_r(x) \forall r \in R)} \quad (11)$$

For our experimentation, we have trained a FRBC. In order to train one, we used a genetic algorithm that optimizes the fuzzy partitions and the antecedents and consequents for each rule. The metric to optimize is the Matthew correlation coefficient (MCC):

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (12)$$

where TP is true positive, TN means true negative, FP is false positive, and FN is false negative.

We also add another condition: in case two subjects performed equally on the fitness metric, we prefer those that did so using fewer rules.

## V. EXPERIMENTATION AND RESULTS

In this section, we evaluate the performance of the proposed framework in the artistic image classification problem for three different settings:

- 1) Author identification using fuzzy rules.
- 2) Classification of type, school, timeframe, and author for paintings using single and multi-task settings.
- 3) Remaining tasks deep features activations using fuzzy rules.

### A. Datasets

For our experimentation, we have used the SemArt dataset [15]. This dataset consists of 21,384 painting images. Following the original data partition in [15], 19,244 images are used for training (i.e., a 90%), 1,069 for validation, and 1,069 for test (i.e., a 5% each). Each painting has an associated textual artistic comment. In this dataset four different classification tasks are proposed:

- Type: each painting is classified according to 10 different common types of paintings: portrait, landscape, religious, etc.
- School: each painting is identified with different schools of art: Italian, Dutch, French, Spanish, etc. There are a total of 25 classes of this kind.
- Timeframe: this attribute, which corresponds to periods of 50 years evenly distributed between 801 and 1900, is used to classify each painting according to its creation date. We consider only the timeframes where more than 10 paintings are present. This corresponds to 18 classes.
- Author: it corresponds to the author of each painting. We consider a total of 350 painters, that comprise the set of authors with more than 10 paintings in the dataset.

We have also used the WikiArt dataset. This dataset is a collection of high-resolution images of artworks and their associated metadata that were scraped from Wikipedia [50]. The WikiArt dataset contains over 81,000 images of fine art paintings representing a wide range of artistic styles and historical periods from the 11th century to the present day. Each image in the dataset is accompanied by a set of metadata, including the title of the artwork, the artist, the year of creation, the medium used, and the dimensions of the artwork, among other attributes.

Both datasets share 1958 paintings, which includes the 10% of the Semart Dataset and 0.02% of the WikiArt dataset. However, the models trained using the WikiArt dataset that we present in the following sections did not train using any of those images to avoid data leakage between the two datasets.

### B. Results for explainable discrimination of particular authors

We will first consider author identification, which is one of the most relevant tasks of artistic curation. Not only to properly identify the original painter of one artistic piece but also to detect possible forgeries or false attributions. Deep learning models can be used to solve this task but they present some relevant problems. First, explanation methods for deep learning models, like the previously discussed Grad-CAM, rely

**Table II:** Results for author and style correlation in Semart. Predictions for each style are generated on a ResNet fine-tuned in the Wikiart dataset. The incorrect style indicates how many times a painting was assigned to a style that does not correspond to the author by the model.

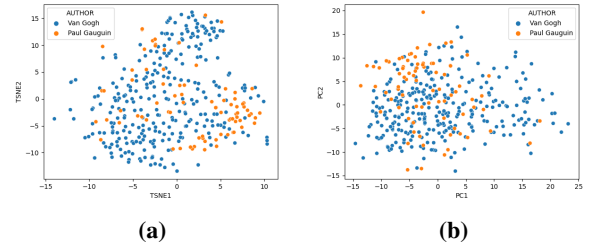
Artist	Number of paintings in Semart	Incorrect style
Albrecht Durer	79	0.25
Camille Pissarro	22	0.00
Childe Hassam	8	0.00
Claude Monet	92	0.03
Edgar Degas	64	0.03
John Singer Sargent	6	0.16
Paul Cezanne	76	0.07
Vincent Van Gogh	291	0.04

on salience maps. This is good enough for visual identification of individual objects in an image, but tasks such as author or school identification do not produce salience maps that are easily interpretable. Secondly, these models output the likelihood of a painting’s authorship, but their predictions tend to be overconfident and unrealistic. Because of these issues, we use a FRBC to solve this problem. A FRBC can distinguish between particular authors of interest using interpretable rules and output a more trustworthy estimation of the reliability of such prediction [51], which solves the problems previously discussed.

As a way of illustrating this application, we shall construct a FRBC to distinguish two painters that were acquainted in real life: Paul Gauguin and Vincent Van Gogh. It is exciting to see if the resulting rules match the actual knowledge that we have of both of them. They are considered post-impressionist painters, with strong similarities and differences in their style.

First, to obtain the style characterization of the SemArt paintings, we apply the style characterization model trained on the WikiArt dataset. We do not expect a significant domain shift between both datasets as the SemArt dataset was not collected with a particular bias in the selection process. However, in order to check how good the performance was in SemArt, we compared the results using the painters that have paintings in both datasets. We measured how often Semart assigned a style to a painting that was different from the styles that Wikiart listed for the painter (Table II). We found a total of 8 common artists in both datasets, with different degrees of misclassification. Of course, some errors are more important than others, i.e., it is not the same to incorrectly classify a pointillist painting as impressionist rather than medieval art. However, the whole complexity of this problem is left open in this work, which is closely related to ordinal classification problems [52]. We found the results satisfactory, as only one painter found significant misclassification: Albert Durer. The rest presented an error of less than 0.10 or included very few paintings (Singer Sargent).

The SemArt dataset contains 291 paintings from Van Gogh and 81 from Gauguin. Fig. 3 shows a PCA and a t-distributed stochastic neighbor embedding (TSNE) visualization of the paintings for each class using our style and Grad-CAM characterization. We derive the rules tuning a genetic algorithm for the FRBC described in Section IV-C. As the number of



**Fig. 3: Visualisation of Van Gogh and Gauguin samples** using TSNE (a) and PCA (b). The features reduced are the style and Grad-CAM features.

**Table III:** Performance for a GBC and our FRBC in the Van Gogh/Gauguin identification task using the relevant features found from a GBC on the whole set of features.

GBC		FRBC	
Accuracy	MCC	Accuracy	MCC
0.88	0.53	0.88	0.62



possible antecedents is too high for the genetic algorithm to obtain a good result, we first trained a Gradient boosting classifier (GBC) [53] using all the input features. This classifier obtained a 100% accuracy but used many features that we know should not be relevant to this task. Then, we computed the feature importance for this classifier and used the seven most important ones to train the FRBC. Table III shows the performance for both the GBC and the FRBC on the selected set of features. We obtained good results for both approaches, which again proves that the style and Grad-CAM heatmap characterization was successful. Indeed, we obtained better results with the FRBC than the GBC in the reduced set of features.

Fig. 4 shows the rules obtained to differentiate both authors and the DS and individual accuracy for each rule in all the training samples they fired. We obtained three successful rules for Van Gogh and one for Gauguin. From those rules, we can see that Synthetic Cubism is a very good feature to identify Van Gogh’s paintings compared to Gauguin, and the Early Renaissance style is the second best. We only found one relevant pattern for Gauguin. One interesting issue is that the best features to discriminate both artists are styles that did not exist in the actual time of the painters. This can indicate that these painters already started some of the traits that characterized those artistic movements.

### C. Results for the classification tasks

We studied the four different classification tasks (type, school, timeframe, and author identification) using the training/test partitions as described in Section V-A. To measure the performance for each task, we have computed the classification accuracy. using The following classification methods are considered:

- 1) The ResNet50, VGG16, Visual Transformer [54] and the ConvNext [55] networks using their corresponding pre-trained weights. We adapt the last layer to match the

Author	Antecedents	DS	Train Acc	Test Acc
	1 IF New Realism IS Low AND Post Impressionism IS Medium	0.0076	0.5000	0.0000
	2 IF Early Renaissance IS Medium AND New Realism IS Medium AND Synthetic Cubism IS Medium	0.0740	0.7777	1.0000
	3 IF Early Renaissance IS Low AND Synthetic Cubism IS High	0.2517	0.9390	0.8888
	4 IF Synthetic Cubism IS Low	0.4624	0.9097	0.9250
	5 IF Contemporary Realism IS Medium AND Synthetic Cubism IS Low AND Relevant area IS Low	0.0092	0.0000	0.0000
	6 IF Contemporary Realism IS Medium AND Minimalism IS Low	0.3389	0.7586	0.7692
	7 IF Early Renaissance IS Medium AND Minimalism IS Medium AND Synthetic Cubism IS Medium	0.0124	0.0000	0.0000

**Fig. 4:** Rules that differentiate Van Gogh from Gauguin paintings. DS stands for Dominance Score. Train and Test acc. determine the percentage of samples where each rule correctly fired in each data partition.

number of target classes. These solutions only consider the visual information for each image.

- 2) The same architectures fine-tuned in an MTL setting for all the different classes, so that context is captured by the shared information between tasks. We retrained all the weights for each network.
- 3) ContextNet: The ResNet50 with information captured from contextual annotations and metadata, using node2vec representations represented by a knowledge graph (KGM) [16].
- 4) Our proposed classification framework in Fig. 1 using the ResNet50 to extract visual features and BoW/TF-IDF and FCM to encode the textual annotations. We use the contextual features as a regularisation element in the training process and append both vectors of features (marked as “append” in Table IV). For the case of the BoW codification, we also test a lighter model that uses only the top 100 most popular words.
- 5) Our proposed classification framework in Fig. 1 uses the ResNet50 to extract visual features and BoW/TF-IDF and FRBC to encode the textual annotations. We use the contextual features as a regularising element in the training process and append both vectors of features (marked as “append” in Table IV).
- 6) Our proposed classification framework in Fig. 1 using the ResNet50 to extract visual features and a CLIP autoencoder to encode the textual annotations. CLIP embeddings are size of 1024.
- 7) The combination of our MTL approach and contextual encodings using FCM and CLIP.

Table IV shows the results for each of the tasks and models. Analyzing from top to bottom, we can see that the MTL methods using visual information alone performed worst in average than those that used MTL or contextual features. Comparing the KGM, K-Means and FCM encodings, the BoW-FCM performed better in “Type” and “Author” tasks, whilst K-Means embeddings did so in the other two. BoW<sub>100</sub>+FCM embeddings obtained the best results in average. When considering contextual information using MTL, KGM, and FCM-based methods, the performance improved substantially for all classes.

Comparing the performance of CLIP features with FCM,

the former ones obtained a best average performance in all the ResNet cases. This is specially relevant as CLIP features are considerable more expensive to compute than the FCM memberships.

Comparing between the FCM models tested, those that used most words for the contextual embedding performed poorly on the Author task, in which the BoW model with only 100 words performed significantly better than the rest of the FCM-based models. It was also the best performing method for this class compared to the KGM and MTL-based proposals. This could be due to the fact that the “Author” classification is the most complicated task, with only a few learning examples per class, and the available contextual vectors are not specific enough to help discriminate in those cases. Appending the contextual vector instead of using it to regularise the gradient seems to have a similar effect on the final performance of the system. Since we are not guaranteed to have textual annotations, it is preferable to use those that only required them in the training process.

We have also joined both paradigms using an MTL model with the two different contextual vectors as a regularisation element. These models outperformed the rest of the models considering the average of all tasks. MTL-FCM and MTL-CLIP obtained an average of 0.661 and 0.662 accuracy, respectively, whilst the second best model, the BoW<sub>100</sub> + FCM, obtained a 0.647. The best existing previous approach, the ContextNet, obtained the fourth best performance with 0.6444 average accuracy on the four tasks.

Finally, we have analyzed the results obtained using the ConvNext and the ViT models. Visual-only features obtained the best results for “Type” and “TimeFrame”, but performed very poorly in the “Author” task. MTL paradigm improved the results in task, but did not in the others. However, it is also important to note that when using MTL we have to train only one model instead of four, which is a significant advantage. The best average result was obtained using MTL with FCM embeddings with the ConvNext architecture.

#### D. Results mapping style characteristics to deep features

We can use a FRBC in the same way as in Section V-B to map painting styles with the activation of internal features the deep learning model used to classify them. First, we studied



**Table IV:** Correct Classification Ratio results for the different attributes on SemArt Dataset test partition (see Section III-A and Section V-C for more details).

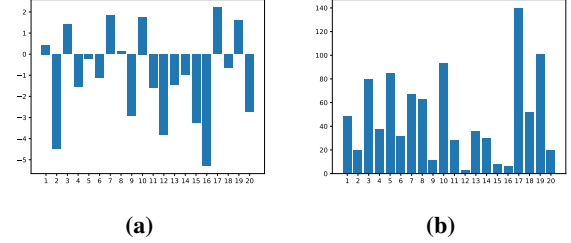
	Method	Architecture	Type	School	TimeFrame	Author	Average
1)	Visual Only	VGG16	0.706	0.502	0.418	0.482	0.527
		ResNet	0.726	0.557	0.456	0.500	0.559
		ViT	0.796	0.673	0.614	0.241	0.581
		ConvNext	<b>0.804</b>	0.680	<b>0.646</b>	0.226	0.589
2)	MTL	VGG16	0.732	0.585	0.497	0.513	0.581
		ResNet	0.763	0.565	0.464	0.431	0.555
		ViT	0.790	0.674	0.651	0.564	0.669
		ConvNext	0.789	0.688	0.605	0.537	0.654
3)	K-Means	ResNet	0.800	0.666	0.628	0.186	0.570
4)	Node2Vec	ResNet	0.786	0.647	0.597	0.548	0.644
5)	ResNet	BoW + FCM	0.794	0.655	0.604	0.238	0.572
		BoW + FCM-append	0.802	0.654	0.584	0.230	0.567
		TF-IDF + FCM	0.786	0.645	0.604	0.229	0.566
		TF-IDF + FCM-append	0.778	0.654	0.589	0.226	0.561
		BoW <sub>100</sub> + FCM	0.792	0.630	0.586	0.559	0.647
6)	ResNet	TF-IDF + FRBC	0.785	0.643	0.597	0.233	0.564
		TF-IDF + FRBC-append	0.759	0.623	0.533	0.154	0.517
7)	CLIP-context	Resnet	0.784	0.649	0.601	0.215	0.560
8)	MTL-FCM	MTL-CLIP	0.790	0.677	0.630	0.551	0.662
		Resnet	<b>0.804</b>	0.691	0.618	0.531	0.661
		ViT	0.796	0.681	0.617	0.562	0.664
		ConvNext	0.793	<b>0.711</b>	0.630	<b>0.568</b>	<b>0.675</b>

which deep features of the penultimate layer are dominant for each sample, denoting as dominant the feature with the highest activation. In order to do so, we have computed the average value and the number of times each feature presented the biggest value in a sample (Fig. 5). Some features are clearly more dominant than others, i.e., 17 and 19, while others are remarkably low, i.e., 12. We have checked the feasibility of this task by visualizing each deep feature studied using PCA (Fig. 6).

Then, we used a GBC to solve the classification task resulting in a 0.22 accuracy. Since Gradient boosting is considered state-of-the-art in the standard tabular classification [56], we can interpret this 0.22 as an upper bound of accuracy for a FRBC. However, we can convert this problem using a One-versus-All scheme. In this way, instead of one multi-class problem, we have 20 binary classification problems. As these problems are heavily imbalanced, we use the MCC to evaluate the results for each feature. They are shown in Table V.

The lack of positive samples for each feature in comparison with the negative ones deeply affects the performance of the GBC. In order to give a more reliable estimation of the performance of the system, we randomly subsampled a balanced partition for each feature (i.e., we perform random oversampling, Table V, column 2). Using these models, we checked the importance that they gave to each style for their predictions. Based on the relevant ones (when the importance value is bigger than the average), we use a FRBC that learns the corresponding rules to map from the input data to the desired class for each deep feature.

Using this model and fuzzy linguistic variables, we can characterize each painting according to the expressiveness of their visual traits. Table VI shows the MCC for the classification of each of the features using a FRBC. As expected from the PCA visualizations in Fig. 6, the performance is very different among the deep features. Finally, in Fig. 7, we show the resulting rules obtained for some of the features where the classification was most successful.



**Fig. 5: Study of deep feature activations.** (a) Average value of the 20 deep features used in MTL-FCM predictions. (b) Histogram containing the number of times that each feature presented the biggest value for each sample in the training set.

**Table V:** Performance measured using MCC for a GBC in the original and a balanced partition obtain by subsampling the original dataset ( $-1$  is the worst possible value and 1 the best).

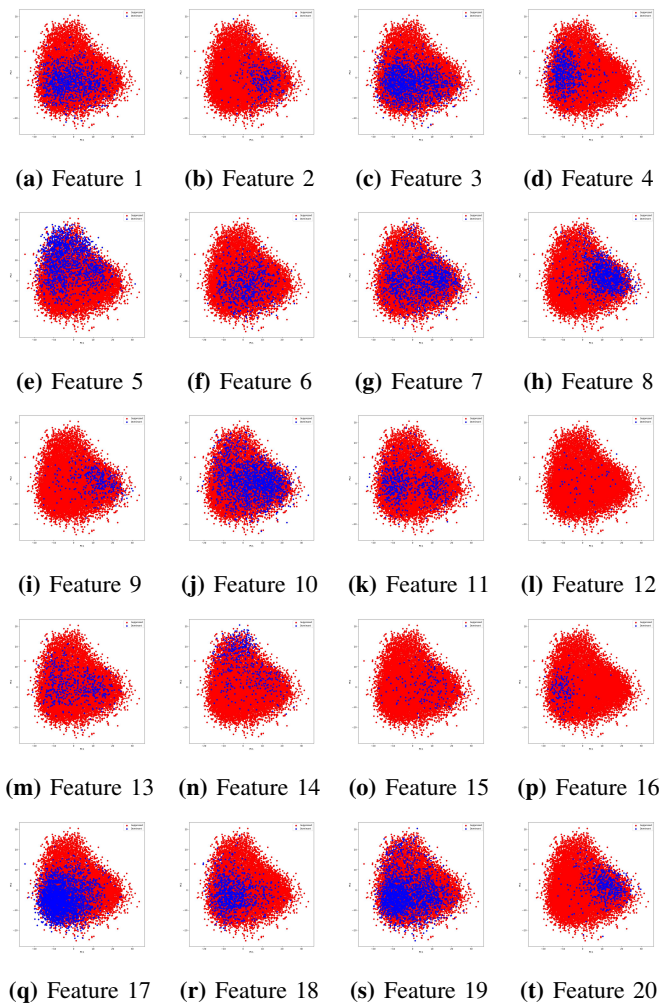
Feature	MCC performance	
	Original partition	Balanced partition
1	0.08	0.26
2	0.00	0.52
3	0.10	0.30
4	0.03	0.43
5	0.23	0.48
6	0.02	0.25
7	0.05	0.25
8	0.06	0.54
9	0.10	0.40
10	0.07	0.21
11	0.07	0.37
12	0.00	0.34
13	0.00	0.18
14	0.20	0.56
15	0.00	0.19
16	0.00	<b>0.60</b>
17	0.22	0.42
18	0.05	0.34
19	0.06	0.20
20	0.00	0.56

## VI. CONCLUSIONS AND FUTURE LINES

In this paper, we have proposed a new method to combine visual features and contextual annotations, using both a fuzzy membership encoding based on the FCM, a FRBC and CLIP features. We used these methods in a classification framework that considers a fine-tuned ResNet50 enriched to extract the visual features from a dataset of artistic images. This frame-

**Table VI:** MCC obtained for each feature using a FRBC.

Feature	MCC	Feature	MCC
1	0.2369	11	0.0000
2	0.5001	12	0.0000
3	0.2076	13	0.0954
4	0.2076	14	0.4666
5	0.4440	15	0.6888
6	0.1611	16	0.4714
7	0.1941	17	0.3708
8	0.4512	18	0.1478
9	0.3612	19	-0.2377
10	0.2432	20	0.4552



**Fig. 6:** PCA projections for the deep features. Blue dots mark samples where the feature is dominant.

work learns to solve a classification problem and to reconstruct the features extracted from the contextual information for each image, which helps the network generalize better, as it does not need to rely only on visual cues to classify each sample. Besides, we have introduced different XAI methods using fuzzy rules to interpret the features and results obtained with these methods.

The comparison between context-aware models with similar visual-only classification frameworks shows favorable results for the former ones, as originally expected. We obtained the best results overall using an MTL paradigm with contextual information. Using fuzzy rules, we also showed how some of the deep features used by the best model can be characterized according to the relevant parts of the image and the style of painting. In addition, we reported how some painters can be successfully distinguished one from another using fuzzy rules and painting styles. As a way of example, we show in this paper a comparison between Paul Gauguin and Vincent Van Gogh

Future lines of our research shall study more expressive features to represent some of the image characteristics [57]

and apply methods to improve the performance of the FRBC in imbalanced datasets. For example, we shall consider the use of information and complexity metrics as features to describe the painting [6]. We also intend to develop a metric that can compute how good is a commentary that describes an image so that the additional information present in the text can be quantified.

## VII. ACKNOWLEDGEMENTS

This work was supported in part by Oracle Cloud credits and related resources provided by Oracle for Research and by MCIN/AEI/10.13039/501100011033 and ERDF “A way of making Europe” under grant CONFIA (PID2021-122916NB-I00).

## REFERENCES

- [1] M. Barni, A. Pelagotti, and A. Piva, “Image processing for the analysis and conservation of paintings: opportunities and challenges,” *IEEE Signal Processing Magazine*, vol. 22, no. 5, pp. 141–144, 2005.
- [2] E. J. Crowley and A. Zisserman, “The art of detection,” in *European Conference on Computer Vision*. Springer, 2016, pp. 721–737.
- [3] A. Lecoutre, B. Negrevergne, and F. Yger, “Recognizing art style automatically in painting with deep learning,” in *Asian conference on machine learning*. PMLR, 2017, pp. 327–342.
- [4] Y. Zeng, Y. Gong, and X. Zeng, “Controllable digital restoration of ancient paintings using convolutional neural network and nearest neighbor,” *Pattern Recognition Letters*, vol. 133, pp. 158–164, 2020.
- [5] G. Carneiro, N. P. Da Silva, A. Del Bue, and J. P. Costeira, “Artistic image classification: An analysis on the printart database,” in *European Conference on Computer Vision*. Springer, 2012, pp. 143–157.
- [6] J. M. Silva, D. Pratas, R. Antunes, S. Matos, and A. J. Pinho, “Automatic analysis of artistic paintings using information-based measures,” *Pattern Recognition*, vol. 114, p. 107864, 2021.
- [7] B. Guo and P. Hao, “Analysis of artistic styles in oil painting using deep-learning features,” in *2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. IEEE, 2020, pp. 1–4.
- [8] W. R. Tan, C. S. Chan, H. E. Aguirre, and K. Tanaka, “Ceci n’est pas une pipe: A deep convolutional network for fine-art paintings classification,” in *2016 IEEE international conference on image processing (ICIP)*. IEEE, 2016, pp. 3703–3707.
- [9] A. Elgammal, B. Liu, D. Kim, M. Elhoseiny, and M. Mazzone, “The shape of art history in the eyes of the machine,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.
- [10] S. M. Anwar, M. Majid, A. Qayyum, M. Awais, M. Alnowami, and M. K. Khan, “Medical image analysis using convolutional neural networks: a review,” *Journal of Medical Systems*, vol. 42, no. 11, pp. 1–13, 2018.
- [11] N. Aloysius and M. Geetha, “A review on deep convolutional neural networks,” in *2017 International Conference on Communication and Signal Processing (ICCCSP)*. IEEE, 2017, pp. 588–592.
- [12] W. Rawat and Z. Wang, “Deep convolutional neural networks for image classification: A comprehensive review,” *Neural Computation*, vol. 29, no. 9, pp. 2352–2449, 2017.
- [13] E. Cetinic, T. Lipic, and S. Grgic, “Fine-tuning convolutional neural networks for fine art classification,” *Expert Systems with Applications*, vol. 114, pp. 107–118, 2018.
- [14] T. E. Lombardi, *The classification of style in fine-art painting*. Pace University, 2005.
- [15] N. Garcia and G. Vogiatzis, “How to read paintings: semantic art understanding with multi-modal retrieval,” in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018.
- [16] N. Garcia, B. Renoust, and Y. Nakashima, “Context-aware embeddings for automatic art analysis,” in *Proceedings of the 2019 on International Conference on Multimedia Retrieval*, 2019, pp. 25–33.
- [17] C. B. E. Vaigh, N. Garcia, B. Renoust, C. Chu, Y. Nakashima, and H. Nagahara, “Gcnboost: Artwork classification by label propagation through a knowledge graph,” *arXiv preprint arXiv:2105.11852*, 2021.
- [18] R. Taber, “Knowledge processing with fuzzy cognitive maps,” *Expert systems with applications*, vol. 2, no. 1, pp. 83–87, 1991.
- [19] X. Chen, S. Jia, and Y. Xiang, “A review: Knowledge reasoning over knowledge graph,” *Expert Systems with Applications*, vol. 141, 2020.

**Fig. 7:** Most important rules (DS >0.1) that identify some of the deep features studied.

Feature 2	DS
IF Early Renaissance IS High AND Northern Renaissance IS High	0.4968
Feature 8	
IF Early Renaissance IS High	0.4332
Feature 15	
IF Cubism IS Low AND Early Renaissance IS High AND Pointillism IS Low	0.1065
IF Early Renaissance IS High AND Rococo IS Low	0.3906
Feature 16	
IF Analytical Cubism IS Low AND Naive Art Primitivism IS Medium AND Pointillism IS Medium	0.2199
IF Contemporary Realism IS High AND Cubism IS Low AND High Renaissance IS Low	0.2932
IF Analytical Cubism IS Low AND Color Field Painting IS Medium AND Pop Art IS Low	0.1334

- [20] M. Newman, *Networks*. Oxford university press, 2018.
- [21] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, oct 2008.
- [22] J. Fumanal-Idocin, A. Alonso-Betanzos, O. Cordón, H. Bustince, and M. Minárová, "Community detection and social network analysis based on the italian wars of the 15th century," *Future Generation Computer Systems*, vol. 113, pp. 25–40, 2020.
- [23] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, p. 814, 2005.
- [24] S. P. Borgatti, A. Mehra, D. J. Brass, and G. Labianca, "Network analysis in the social sciences," *Science*, vol. 323, no. 5916, pp. 892–895, 2009.
- [25] J. Fumanal-Idocin, O. Cordón, G. P. Dimuro, A.-F. R. López-de Hierro, and H. Bustince, "Quantifying external information in social network analysis: An application to comparative mythology," *IEEE Transactions on Cybernetics*, 2023, in press.
- [26] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.
- [27] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.
- [28] S. Targ, D. Almeida, and K. Lyman, "Resnet in resnet: Generalizing residual architectures," *arXiv preprint arXiv:1603.08029*, 2016.
- [29] N. Gonthier, Y. Gousseau, S. Ladjal, and O. Bonfait, "Weakly supervised object detection in artworks," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, pp. 1–18.
- [30] V. Tonkes and M. Sabatelli, "How well do vision transformers (vts) transfer to the non-natural image domain? an empirical study involving art classification," *arXiv preprint arXiv:2208.04693*, 2022.
- [31] N. Garcia, B. Renoust, and Y. Nakashima, "Contextnet: representation and exploration for painting classification and retrieval in context," *International Journal of Multimedia Information Retrieval*, vol. 9, no. 1, pp. 17–30, 2020.
- [32] J. Fumanal-Idocin, Z. Takáč, L. Horanská, H. Bustince, and O. Cordon, "Fuzzy clustering to encode contextual information in artistic image classification," in *Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Springer International Publishing, 2022, pp. 355–366.
- [33] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy c-means clustering algorithm," *Computers & Geosciences*, vol. 10, no. 2-3, pp. 191–203, 1984.
- [34] M. V. Conde and K. Turgutlu, "Clip-art: contrastive pre-training for fine-grained art classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3956–3960.
- [35] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8748–8763.
- [36] O. Cordón, M. J. Del Jesus, and F. Herrera, "A proposal on reasoning methods in fuzzy rule-based classification systems," *International Journal of Approximate Reasoning*, vol. 20, no. 1, pp. 21–45, 1999.
- [37] O. Cordón, F. Herrera, and P. Villar, "Analysis and guidelines to obtain a good uniform fuzzy partition granularity for fuzzy rule-based systems using simulated annealing," *International Journal of Approximate Reasoning*, vol. 25, no. 3, pp. 187–215, 2000.
- [38] J. A. Sanz, A. Fernández, H. Bustince, and F. Herrera, "Ivturs: A linguistic fuzzy rule-based classification system based on a new interval-valued fuzzy reasoning method with tuning and rule selection," *IEEE Transactions on Fuzzy Systems*, vol. 21, no. 3, pp. 399–411, 2013.
- [39] O. Cordón *et al.*, *Genetic fuzzy systems: evolutionary tuning and learning of fuzzy knowledge bases*. World Scientific, 2001, vol. 19.
- [40] E. G. Mansoori, "FRBC: A fuzzy rule-based clustering algorithm," *IEEE Transactions on Fuzzy Systems*, vol. 19, no. 5, pp. 960–971, 2011.
- [41] J. M. Mendel and P. P. Bonissone, "Critical thinking about explainable ai (xai) for rule-based fuzzy systems," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 12, pp. 3579–3593, 2021.
- [42] F. S. Khan, S. Beigpour, J. Van de Weijer, and M. Felsberg, "Painting-91: a large scale database for computational painting categorization," *Machine Vision and Applications*, vol. 25, no. 6, pp. 1385–1397, 2014.
- [43] G. Castellano, V. Digeno, G. Sansaro, and G. Vessio, "Leveraging knowledge graphs and deep learning for automatic art analysis," *Knowledge-Based Systems*, vol. 248, 2022.
- [44] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.
- [45] G. Castellano, G. Sansaro, and G. Vessio, "Integrating contextual knowledge to visual features for fine art classification," *arXiv preprint arXiv:2105.15028*, 2021.
- [46] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.
- [47] N. Pal, K. Pal, J. Keller, and J. Bezdek, "A possibilistic fuzzy c-means clustering algorithm," *IEEE Transactions on Fuzzy Systems*, vol. 13, no. 4, pp. 517–530, 2005.
- [48] M. Kiani, J. Andreu, and H. Hagrais, "A temporal type-2 fuzzy system for time-dependent explainable artificial intelligence," *IEEE Transactions on Artificial Intelligence*, 2022, in press.
- [49] J. Andreu-Perez, L. L. Emberson, M. Kiani, M. L. Filippetti, H. Hagrais, and S. Rigato, "Explainable artificial intelligence based analysis for interpreting infant fnirs data in developmental cognitive neuroscience," *Communications biology*, vol. 4, no. 1, p. 1077, 2021.
- [50] B. Saleh and A. Elgammal, "Large-scale classification of fine-art paintings: Learning the right metric on the right feature," *arXiv preprint arXiv:1505.00855*, 2015.
- [51] G. Fernández, J. Aledo, J. Gamez, and J. Puerta, "Factual and counterfactual explanations in fuzzy classification trees," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 12, pp. 5484–5495, 2022.
- [52] P. Bellmann and F. Schwenker, "Ordinal classification: Working definition and detection of ordinal structures," *IEEE Access*, vol. 8, pp. 164 380–164 391, 2020.

- [53] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, R. Mitchell, I. Cano, T. Zhou *et al.*, “Xgboost: extreme gradient boosting,” *R package version 0.4-2*, vol. 1, no. 4, pp. 1–4, 2015.
- [54] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, “An image is worth 16x16 words: Transformers for image recognition at scale,” *arXiv preprint arXiv:2010.11929*, 2020.
- [55] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, “A convnet for the 2020s,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022*, pp. 11 976–11 986.
- [56] J. Fumanal-Idocin, O. Cerdón, and H. Bustince, “The krypteia ensemble: Designing classifier ensembles using an ancient spartan military tradition,” *Information Fusion*, vol. 90, pp. 283–297, 2023.
- [57] J. Machajdik and A. Hanbury, “Affective image classification using features inspired by psychology and art theory,” in *Proceedings of the 18th ACM International Conference on Multimedia*, ser. MM ’10. New York, NY, USA: Association for Computing Machinery, 2010, p. 83–92.